DOCUMENT RESUME

ED 077 245 EM 011 173

AUTHOR Kaplow, Roy; And Others

TITLE Illustrations of Conversational, Inquiry, Problem-Solving, and Questionnaire Type Interactions Within The TICS System.

INSTITUTION Massachusetts Inst. of Tech., Cambridge. Dept. of Metallurgy and Materials Science.

SPONS AGENCY National Science Foundation, Washington, D.C. Office of Computing Activities.

PUB DATE Mar 73

NOTE 5p.; Proceedings, Seventh Annual Princeton Conference on Information Sciences and Systems, March 22-23, 1973

EDRS PRICE MF-$0.65 HC-$3.29

DESCRIPTORS *Classification; Computer Assisted Instruction; *Computers; *Computer Science; Data Bases; Instructional Media; *Interaction; Man Machine Systems; State of the Art Reviews

IDENTIFIERS TICS System

ABSTRACT

It is possible to separate the use of computers for instructional purposes into four modes, distinguished by the style of user interaction and sometimes by the implementation methods used. The four major modes are: 1) conversational, in which the interaction resembles an exchange between two persons; 2) data-based, wherein one component of the man-machine system extracts from the other information useful to the solution of a problem; 3) computational, in which the user focuses upon the computational power of the computer; and 4) drill, a repetitive, structurally fixed interaction in which only the specific contents are changed. Classifications of this sort overemphasize difference between styles of use, but attending to structural and pedagogical attributes has its uses. It provides a framework for characterizing system capabilities and applications, a shorthand for communication among computer scientists, and guidelines for novice authors of instructional material. (PB)

FILMED FROM BEST AVAILABLE COPY

Proceedings, Seventh Annual Princeton Conference on Information Sciences and Systems, March 22-23, 1973

ED 077245

# ILLUSTRATIONS OF CONVERSATIONAL, INQUIRY, PROBLEM-SOLVING, AND QUESTIONNAIRE TYPE INTERACTIONS WITHIN THE TICS SYSTEM*

U S DEPARTMENT OF HEALTH EDUCATION & WELFARE NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

by

Roy Kaplow, Samuel H. Desch, Jr., David O. Pettijohn, Melvin H. Rodman, and Franklin C. Smith, Jr.

Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

It is possible to divide the use of computers for instructional purposes into a number of categories or modes. We recognize that this division overemphasizes the differences between styles of use, and that it contributes to furthering an imprecise jargon. Nonetheless, it seems worthwhile to stress, in this way, the structural and pedagogical attributes which do stand out. This can provide a framework for characterizing system capabilities and applications, a shorthand for communication among authors, and guidelines for inexperienced authors of instructional material. Following, in part, divisions drawn in a recent Carnegie Commission and Rand Corporation Study[1], we list the following categories: Conversational (tutorial and socratic); Data-based (enquiry, questionnaire and generative); Computational (problem-solving, simulation and gaming); and Drill. These categories are distinguished from one another in the style of user interaction developed and sometimes by the implementation methods used.

## Interaction Modes

### Conversational

This type of interaction is programmed to have the appearance of a conversation between two persons. In practice, the conversation is often designed such that the student's input is, or is expected to be, rather terse, to minimize the computer's problems in understanding language. For the conversational flow to be maintained, the student's input must have been anticipated, not simply that it may be recognized, but so that the program can respond meaningfully.

In principle, the actual flow of the conversation is responsive to the student's input, and different students may see different interactions. In practice, many programs do not embody very complex structural patterns; often, all students pass through a lineal network, likened to beads-on-a-string, in which the only diversity is in the short paths within the beads.

The conversational mode of presentation may be further divided into two subcategories, tutorial and socratic. In the former, the tendency is for the computer to present information, ask a question, get a student response, analyze the response, and then to take response-dependent actions. These actions might include calling subroutines, giving hints and selecting paths along multi-choice branches. With socratic-type programs, the balance of the conversation is shifted; the student queries the computer, the computer answers and the student takes response dependent actions such as asking other questions, making declarative statements, and requesting jumps to other parts of the program. An important characteristic of the conversational mode (of both types) is the notion that the program actively guides the student, pointing out errors and giving advice relative to the current context.

### Data-based

In the enquiry type of interaction, the student, working in the framework of a given problem or question, can ask the computer for information and data which he believes will help him towards an answer, or towards a more complete understanding. The information and data is stored as a data-base within the computer's memory and the student is given appropriate information-retrieval or "look-up" commands to use. Ordinarily, one thinks of the accessed data as being fixed during the time of its use by a student, but that is not a fundamental requirement. In the pure form of enquiry mode, the retrieval programs operate without reference to the specific, immediate context in which the information is sought. This may be contrasted to what we have termed the socratic conversational interaction, above, in which the student's questions or the responses to them may be interpretable only in the context of the immediately preceding statements. One of the interesting aspects of enquiry mode interaction, therefore, is that it can be superimposed readily on other forms of interaction as a student-controlled interruption, since it is independent of the specific details which govern the program logic and flow.

The questionnaire mode, as the name implies, is an interaction in which the computer collects information from the user, by asking direct questions. It is useful to think of this mode as the complement of enquiry mode, that is, as a process of filling in a data base, bit by bit (even though the data base need not be real). In contrast to tutorial mode, there is no emphasis on the computer's conveying information through statements, the questions are less likely to require deep analysis, and the flow is probably less conversational; nonetheless, the questions asked may depend on earlier responses. Although one usually thinks of a computerized questionnaire simply as a professionally relevant tool (e.g., for obtaining medical histories) the mode may also be applied to teaching. It might be useful, for example, for medical students, nurses, or aides in training to

* Research supported by the National Science Foundation, Office of Computing Activities.

EM 011 173

play the roles of different incoming patients, seeing both the kind of information thought to be important under various circumstances and the tentative diagnosis reached, if any.

Of the various modes, questionnaire is the one most readily associated with testing in educational applications. It should be stressed, therefore, that the latter function can also be served--possibly better--with other types of interaction.

A generative interaction system is distinguished less, perhaps, by the flavor of the interaction--as perceived by the student--than it is by the structure of the enabling computer programs. A generative system utilizes a data-base which describes the subject matter to be taught; the data-base contains references to each of the items with which one deals in that subject, and descriptions or procedures for the relationships which relate the items to one another and the concepts which govern their properties and behavior. Ideally, the data base is structured so as to be quite general, with its stored items readily manipulated and interrogated by appropriate routines. A generative system, including natural language handling capabilities, might aim: a) to construct the data base, dynamically, from a series of declarative input statements from the author, or, perhaps, from the author's responses to direct questions. b) to construct questions and problems, based on the data base, to pose to the student. c) to test students' declarative statements by referring to the data base. d) to answer students' questions relative to items in the data base.

These are difficult capabilities to achieve with any generality and, as yet, applications have been made only to specific topics of limited complexity and have addressed only some of the above mentioned aims.

Computational

In the problem-solving mode, the student is focused mainly on using the computational power of the computer. He may have access to specific-problem oriented subroutines or to a quite general system for mathematical analysis (numerical analysis, most likely). With the former, the student's questions can be specific (e.g., Here are experimental diffraction data, what is the crystal lattice and the lattice dimensions?) With the latter, he must build his own solution out of macroscopic operators (e.g., Here is the data, subtract "the following function", take the Fourier sine transform, convolve that with the "following function", plot the result). At still another level, he may have access to programming languages, such as Fortran, PL/1, LOGO, and Basic, or to an electronic desk-calculator type facility.

Ordinarily, one thinks of the problem-solving mode in connection with giving students access to difficult, professionally relevant problems, and problems whose solutions may be needed for a larger context. Indeed, a good instructional problem-solving facility should be useful to practicing professionals. It is, therefore, perhaps not superfluous to mention that the experience gained in modeling, detailing and setting up algorithms for problem solutions is often of greater relevance, for particular stages of a student's education, than the use of those problem solutions in a larger context.

Interactions of the simulation type are built around computer programs which model some aspect of the real world (or of a hypothetical one). The student may supply and vary a number of input parameters (independent variables) on which the predictions of the model depend. The computer then presents the student with the resulting picture, description, or derived output parameter values for whatever object or process is being modeled. Obvious applications of this mode include not only the simulation of ordinary laboratory experiments but also of very large systems (e.g., a steel plant) and of phenomena that are inherently difficult to see (e.g., the atom displacements around a dislocation defect in a crystal).

In a gaming form of interaction, the student is engaged in a contest, either against an opponent (another student, or the computer) or against ideal or relative standards. The program may have fixed rules for the game and perfect strategies (e.g., tic-tac-toe) or fixed rules and imperfect strategies (e.g., chess). It may, on the other hand, depend not so much on rules as on a simulation model, and the strategies, if any, may be entirely the student's own (e.g., "Design a four element alloy which most nearly satisfies the following requirements: ... ").

Drill

A drill is a structurally fixed interaction repeated over and over again, while only the specific contents are changed. The drill mode has the advantage, for authors, that it allows maximum advantage to be taken of a single, short but well designed format for an interaction. In principle, that format might include any of the other modes of use, and the overall interaction can be interesting and effective in spite of its repetitiveness.

## Illustrations

Examples of the interaction-types might be drawn from a number of sources, but the illustrations given in the remainder of the paper will all be drawn from programs implemented in the TICS system,[2] described in the previous paper. With the exception of the generative system approach, all of the modes mentioned earlier can be implemented within the TICS framework, but the examples here include only tutorial, socratic, enquiry, questionnaire, and problem-solving. It will be noted, of course, that different modes are mixed, even within the limited samples.

The first example is selected from a tutorial-type program on the physics and mathematics of the wave equation.[3] The program is intended to be for M.I.T. undergraduates; the portion illustrated is a slightly modified extraction from a student's interaction lasting about 10-12 hours over 5 separate sessions. The computer always begins its output lines with a colon, and signals when it is ready for

a student response by typing a sequential response number, such as: >17>. The stop/go convention (response 8) and the student's use of "?" (to mean, in effect, "I don't know, but tell me more if you can.") are author-defined conventions, explained to the student when he starts the use of this program. (In all of the illustrations, three vertical dots are used to indicate that part of the interaction has been left out.)

```
: Lets solve the wave eqn by making a change of variables,
: from (x,t) to (q,u) in the eqn; i.e., we want to
: substitute the variables
:                    q = x - vt
:  and
:                    u = x + vt
:  into the eqn
:          Dxx(PSI(x,t)) = (1/vsq)*Dtt(PSI(x,t)).
:  What is the resulting eqn for PSI(q,u)?

>7>?

:  We'll start by deriving Dxx(PSI): The first step is to
: use the chain rule for partial derivs:
:                    Dx(PSI) = Dq(PSI)*Dx(q) + Du(PSI)*Dx(u)
: Next, notice that Dx(q) = 1 and Dx(u) = 1, so
:                    Dx(PSI) = Dq(PSI) + Du(PSI) ...(w1-4).

: We want Dxx(PSI); using eqn (w1-4),
:          Dxx(PSI) = Dx(Dx(PSI))
:                   = (Dq + Du)(Dq(PSI) + Du(PSI))
:                   = Dqq(PSI) + Duu(PSI) + 2Dqu(PSI)
:                                              ...(w1-5).
: This treatment has assumed that PSI is sufficiently
: continuous so that Dqu(PSI)=Duq(PSI).
:  stop.

>8>go on

:  For the other half of the wave eqn, (1/vsq)*Dtt(PSI),
: one can use the same procedure as for Dxx(PSI).
:    First, calculate (1/v)*Dt(PSI) in terms of q and u.
: Just type the answer.

>9>Du(PSI)-Dq(PSI)

: No, here's a hint:  Dt(q)=-v and Dt(u)=v.

>10>Dq(PSI)-Du(PSI)

: good

:  Similar to the way (w1-5) was derived from (w1-4), now
: calculate (1/vsq)Dtt(PSI).  Just type your answer.

>11>(1/vsq)Dtt(PSI)=Dqq(PSI)+Duu(PSI)

: There is another term in the answer.

>12>Dqq(PSI)+Duu(PSI)-2Dqu(PSI)

: good

                              :
                              :

:  By a change of variables, the wave eqn for PSI(x,t) has
: been reduced to the eqn
:                    Dqu(PSI(q,u)) = 0              ...(w1-7).
:  What is the solution PSI to this eqn?

>18>(c_1*q)+(c_2*u)

: Close; your answer is a correct particular solution, but
: I would like a general solution.

                              :
                              :
```

```
:   The change of variables from (x,t) to (q,u) has enabled
: us to get a general functional form for the solution PSI:
:                    PSI(q,u) = f(q) + g(u)          ...(w1-8).
:   What physical interpretation would you give to this
: solution?

>21>moving wave

: If we use (w1-2) and (w1-3) to change back into (x,t), we
: get
:                    PSI(x,t) = f(x-vt) + g(x+vt).
:   What behavior does this represent?

>22>wave moving along a straight line

: yes, but I want one statement containing how many
: traveling waves and what direction they are moving in.

>23>one wave moving in either the + or - x direction

: But f(x-vt) and g(x+vt) each represent one wave moving in
: a given direction.  What then does PSI(x,t) represent?

>24>one wave moving in the +x direction and one wave moving
in the -x direction

: right                       .
                              .
                              .
```

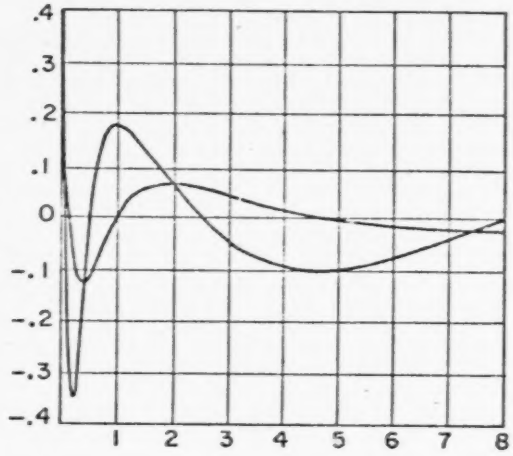
The second example is from a program for aiding students to learn to read scientific German.[4] The entire interaction shown is contained within two TICS nodes. The first of these (extending to student response number 35) presents a rather long German sentence, about which it gives the student information, at his request. All of the student responses which begin with an asterisk are "interruptive" (note that they are not counted in the response numbering). In this instance, these requests are all for information from three different "dictionaries" containing entries for the meaning, etymology, and phonetic pronunciation of words. The other student responses here also follow author-defined conventions, but are specifically relevant to the given sentence. A major portion of this program consists of a large set of similar pairs of nodes, comprising a "drill-and-practice".

```
: Dabei ergab sich u.a. bei der Untersuchung des Radiums
: eine u'berraschend gute Mo'glichkeit, sein
: Begleitelement Barium abzutrennen, das hierbei in
: inactiver Form vorlag, und in einer Ausbeute bis zu
: 80-90% Radiumfrei erhalten werden kann.

>31>verb

: ___ergab sich___(vorlag),___(erhalten werden kann).

>32>v-forms

: Ergab is the past tense of
:
:         ergeben = to yield, deliver up, result in
:                 < er- = up + geben = to give
:
: ich ergab, er ergab, sie ergaben.
: past participle = ergeben (the p.p. of verbs with
                              .
                              .
                              .
```

```
: kann = can, < ko'nnen = to be able.
: ich kann, er kann, sie ko'nnen.

>33>*meaning Begleit.

: begleiten = to escort, go with
                              .
                              .
                              .
>33>*ety Ausbeute.

: Ausbeute = gain, yield, profit
:          < ausbeuten = to exploit
:                      < aus = out +
:                        die Beute = booty

>33>*pro Ausbeute.

: Ausbeute = *owss-beut-uh

>33>w-trans

: There-by yielded itself among others by the
: under-searching of-the radium an over-rushing-ly
: (surprisingly) good possibility, its co-element barium
: away-to-separate, that here-by in inactive form
: in-front-lay, and in an out-booty (yield) up-to to
: 80-90% radium-free obtained become can.

>34>e-trans

: In this connection, the investigation of radium, among
: other elements, yielded a surprisingly good opportunity
: to separate out its co-element barium, which was present
: (here) in inactive form and can be obtained in a yield
: of 80-90% radium-free.

>35>go

: Note the use of "u'berraschend" as an adverb (modifying
: gute = good: surprisingly good), another example of
: German using adjectives as adverbs.
:
: Ergab is the past tense of the strong verb ergeben < er
: + geben, and is declined, of course, like geben.  For
: more on strong verbs (if you're interested) see grammar
: sections 23, 42-44, and 52.
: (Vorleg < vor + liegen is another example of the past
: tense of a strong verb (liegen = to lie)).
:
: Type the English for this sentence:
:
:         Wir erhielten Barium in einer Ausbeute bis zu
:         90% Radiumfrei.

>36>*meaning erhielten


: erhielten = past tense of erhalten = to get, obtain

>36>We obtained barium in a yield of up to 90% radium-free.

: Right.
:
: Here is the next sentence.
```

The third example is drawn from a program dealing with electron wave functions and energies in an approximate model (Wigner-Seitz).[5] The student calculates equilibrium ion distances for a sufficient number of trial valence electron energies to plot an energy versus distance curve, which is then used to derive the theoretical lattice parameter, compressibility and cohesive energy of a solid. In the portion shown, the student is being led by the hand through the calculation for his first trial energy. Thereafter, the student can call the specific computational routines as he chooses, using the interruptive access to supplied subroutines and subsystems, illustrated for the subsystem named calc.

```
: please type your first trial energy (in Rydbergs).

>31>-0.1

: Now that we have chosen an energy, what do we need to do?

>32>integrate the radial equation

: Right.  The computer will now do this.
: The integration is complete.
: What is the next step?

>33>work on wave function

: The next step is to differentiate the radial function.
: The computer will now do this
: This is a plot of dp/dr and p/r against r.  Can you find
: the two sphere radii?
```




```
>34>yes

: Please type your values on one line, separated by a blank.

>35>0.5 2.0

: Your radius 0.5 is much too small.  Only the crossing
: points beyond the second zero of the radial function are
: significant.
: Besides, your energy is higher that the energy of the 1s
: electron in a free sodium atom.

: Please type another energy
                              .
                              .
                              .
: In terms of the sphere radius rs, what is the formula for
: the Fermi energy?

>44>2.21/rs**2

: You now need to calculate the Fermi energies for your
: energy and sphere radii.  The "calc" request is available
: to do this with, or you can do the arithmetic yourself if
: you like.  Would you like instruction in the use of calc?

>45>no
```

```
: Calculate the total energies for your sphere radii and
: type them in

>46>*calc

-.45+2.21/2.24**2
: = -1.0722E-2
-.45+2.21/6.61**2
: = -0.3992
Q
>46>-1.073e-2, -.395

                    .
                    .
                    .
: You can carry out the solution for the next energy
: yourself.  Would you like instruction in how to do this?

>50>yes

: To do the calculation yourself, you need three commands.

: 1) "*sodium" gets a solution of the radial equation.
: You will be asked to type your energy.

: 2) "*boundary" differentiates the radial function.

: 3) "*plot" gives plots of functions occurring in the
: computation.  You will be asked what functions you want
: plotted.  Functions that can be plotted are p, p/r, dp/dr,
: and V.
                    .
                    .
                    .
```

The last example is from a medical history questionnaire.[6] Hidden behind the visible portion of the interaction is the writing of entries about the medicines taken into a report file, and the setting of variables or flags used elsewhere in the questionnaire to affect the questions asked, tentative diagnoses made (if any) and detection of possible emergency situations. The particular question shown illustrates a difficult aspect of questionnaires, namely a situation in which there is a virtually infinite set of potential answers; generally only a small set of these need to be noted specifically (because they affect the rest of the questionnaire), but none can be ignored entirely. In the present situation the author handles the problem by specifically "anticipating" only those medicines which affect the program's logic; this is backed up by using a general purpose subroutine which extracts text items from the list which can then be used individually in outputs and reports (albeit without understanding).

```
                    .
                    .
                    .
: Do you regularly take or use any medicines?  For example
: aspirin, vitamins, birth-control pills, hormones,
: tranquilizers, weight pills, digitalis, fluid pill or
: diuretics, etc.?  Don't forget to count salves and
: ointments

>31>sometimes I do

: Please list all the medicines you are taking.  Type the
: name if you know it (don't worry about spelling) or the
: reason for taking it.  Use a comma to separate the
: medicines on the list.
:
: For example: digoxin, fluid pills, acne pills, vitamines.

>32>ortho novum, insulin, acne medicine

: How long have you been taking ortho novum?

>33>three years

: How long have you been taking insulin?

>34>12 years

                    .
                    .
                    .
: Have you-during the past two weeks-had any of the
: following experiences?

: a) Feeling very drowsy or sleepy-even though you've slept
: long and well
: b) A squeezing or pressure-like pain in the chest
: c) A seizure-a fit or convulsion
: d) Fainting or "passing out"
: e) ...No, I haven't had anything like that in the past
: two weeks...
: Please choose one or more of the above responses by
: letter, separated by comma

>69>a

: Your doctor will be concerned about your being drowsy or
: sleepy, since you are taking insulin.  Rather than
: answering any more questions now, please call the aide
: and ask him to type "advice".  He'll make sure that you
: see the doctor soon.

>70>advice

: This patient should see a doctor today.  If there's time,
: get a blood sugar and a urine test for sugar and acetone
: immediately.  Be sure the doctor gets the report of this
: interview.
```

## References


1. R. E. Levien et al., The Emerging Technology, Carnegie Commission on Higher Education and Rand Corporation Study (1972).

2. Roy Kaplow, D. S. Schneider, F. C. Smith, Jr., and W. R. Stensrud, TICS: The Author Language and Instruction Manual, Massachusetts Institute of Technology (1971); TICS, A System for the Authoring and Delivery of Interactive Instructional Programs, Seventh Annual Princeton Conference (1973).

3. D. O. Pettijohn, Use of the TICS System to Write a Computer Tutorial on the Wave Equation, S. M. Thesis, M.I.T. Department of Metallurgy and Materials Science (1972).

4. S. H. Desch, Jr., An Interactive Computer Aid to Reading Scientific German, M.I.T. Department of Metallurgy and Materials Science (1973).

5. F. C. Smith, Jr., An Interactive Computer Tutorial on the Wigner-Seitz Problem, M.I.T. Department of Metallurgy and Materials Science (1973).

6. M. Rodman, ~~Medical~~ a Medical Questionnaire, Medical Department, M.I.T.